UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/608,888	06/30/2000	Ashwani Garg	19659.01800	1969

7590 05/05/2004

Adam H Tachner
Crosby Heafey Roach & May
PO Box 7936
San Francisco, CA 94120-7936

EXAMINER	
JACOBS, LASHONDA T	

ART UNIT	PAPER NUMBER
2157	8

DATE MAILED: 05/05/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application 09/608,888	Applicant(s) GARG ET AL.	
	Examiner LaShonda T. Jacobs	Art Unit 2157	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *30 June 2000*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-99* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-99* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

Response to Amendment

This Office Action is in response to Applicants' Request for Reconsideration filed on February 9, 2004. Claims 1-99 are presented for further examination.

Claim Rejections - 35 USC § 102

1. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

2. Claims **78-81**, **83-85**, **87-89**, **91-92** and **94-99** are rejected under 35 U.S.C. 102(e) as being anticipated by Wyld et al (hereinafter, "Wyld", 5,978,933).

As per claim **78**, Wyld discloses a fault tolerant, distributed processing, computer apparatus for use in systems, the apparatus comprising:

- a plurality of processes, executing on at least one processor (col. 6, lines 33-48);
- said processes executing an application in the same mode as at least one other application or in a mode different from said one other application, said same and different modes being:

  a) a pure distributed mode where an application is distributed among said processes in an active condition (col. 6, lines 33-48 and col. 7, lines 10-34)

b) a pure fault-tolerant mode where an application executes in at least one process in an active condition and in at least one process in a standby condition (col. 6, lines 33-48 and col. 7, lines 10-34); and

c) a distributed fault-tolerant mode where an application is distributed on multiple processes in an active condition and on at least one process in a standby condition (col. 6, lines 33-48 and col. 7, lines 10-34).

As per claim **79**, Wyld discloses a method in a computer apparatus for fault tolerant and distributed processing of at least one application in a plurality of processes running on at least one processor, the method comprising the steps of:

- executing said application in a distributed fault tolerant mode wherein said application is distributed in an active condition among more than one process and is in standby condition on at least one said process on said processors (col. 6, lines 33-48 and col. 7, lines 10-34);
- providing a plurality of resource sets as units of distribution of said application (col. 4, lines 22-30); and
- a master critical resource set modifying shared data in said application and updating to a shadow resource set of said application on said processes and an active non-critical resource set modifying private data in said application and updating to a standby resource set of said application on another said process (col. 4, lines 22-30 and col. 6, lines 44-65).

As per claim **87**, Wyld discloses a method in a computer apparatus for distributed processing of at least one application in a plurality of processes running on at least one processor; the method comprising the steps of:

- executing said application in a pure distributed mode wherein said application is distributed in an active condition among more than one process (col. 6, lines 33-48 and col. 7, lines 10-34);
- providing a plurality of resource sets as units of distribution of said application (col. 4, lines 22-30); and
- a master critical resource set modifying shared data in said application and updating to a shadow resource set of said application on said processes and an active non-critical resource set modifying private data in said application (col. 4, lines 22-30 and col. 6, lines 44-65).

As per claim **94**, Wyld discloses a method in a computer apparatus for fault tolerant processing of at least one application in a plurality of processes running on at least one processor; the method comprising the steps of:

- executing said application in a fault tolerant mode wherein said application is in an active condition on one process and is in standby condition on another said process on said processors (col. 6, lines 33-48 and col. 7, lines 10-34);
- representing said application by a single resource set (col. 4, lines 22-30); and
- an active single resource set modifying private data in said application and updating to a standby resource set of said application on another said process (col. 4, lines 22-30 and col. 6, lines 44-65).

As per claims **80** and **95**, Wyld further discloses the steps of:

- bringing said resource sets into either of active or standby state on said processes (col. 4, lines 22-30 and col. 6, lines 44-65); and
- said active resource set processing input events and sending update information to said standby resource set (col. 4, lines 22-30 and col. 6, lines 44-65).

As per claims **81** and **96**, Wyld further discloses the step of:

- using a warm start procedure to bring said resource sets into standby state from out of service state (col. 4, lines 22-30 and col. 6, lines 44-65).

As per claims **83**, **91** and **97**, Wyld further discloses the step of:

- providing communication between said application and other applications independent of application location and carrying out said communication external to the application by routing an event to the process where a mapped resource set is active (col. 4, lines 22-30 and col. 6, lines 44-65).

As per claims **84** and **98**, Wyld further discloses the step of:

- transparently sending update messages from said active resource set to a corresponding said standby resource set by performing routing external to said application and routing messages to the process where the resource set is standby (col. 4, lines 22-30 and col. 6, lines 44-65).

As per claims **85** and **99**, Wyld further discloses the steps of:

- bringing said standby resource sets into the active state for recovering from a failure of active resource sets and routing events to new active resource sets (col. 4, lines 22-30 and col. 6, lines 44-65).

As per claim **88**, Wyld further discloses the steps of:

- bringing said resource sets into active state on said processes; and (col. 4, lines 22-30 and col. 6, lines 44-65);
- said active resource set processing input events (col. 4, lines 22-30 and col. 6, lines 44-65).

As per claim **89**, Wyld further disclose the step of:

- using a warm start procedure to bring said resource sets into shadow state from out of service state (col. 4, lines 22-30 and col. 6, lines 44-65).

As per claim **92**, Wyld further discloses the step:

- transparently sending update messages from said active resource set to a corresponding said shadow resource sets by performing routing external to said application and routing messages to the processes where a resource set is shadow (col. 4, lines 22-30 and col. 6, lines 44-65).

Claim Rejections - 35 USC § 103

3. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

4. Claims **1-77**, **82**, **86**, **90** and **93** are rejected under 35 U.S.C. 103(a) as being unpatentable over Wyld in view of Rathunde.

As per claim 1, Wyld discloses a distributed processing computer apparatus for use in systems, the apparatus comprising:

- a plurality of processes executing on at least one processor (col. 6, lines 33-48);
- at least one application executing in a pure distributed mode where said application is distributed in an active condition among more than one of said processes on said processors (col. 6, lines 33-48 and col. 7, lines 10-34);
- a router for providing communications between at least one said application and other applications independent of application locations (col. 6, lines 60-65).
- an ADSM for providing distributed functionality in said application (col. 8, lines 35-63); and
- an ALDM for distributing incoming events to said application (col. 8, lines 35-63).

However, Wyld does not explicitly disclose:

- a system controller for controlling system activation and initial load distribution.

Rathunde discloses a system and method for performing dynamic load balancing during system operation between a pair of application processes including:

- a system controller for controlling system activation and initial load distribution (col. 8, lines 17-32, col. 9, lines 19-26 and lines 37-48).

Given the teaching of Rathunde, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify Wyld incorporating or implementing a load balancing technique in order to improve the load sharing among the application processors by balancing incoming messages/packets across the processors.

As per claim **28**, Wyld discloses a fault tolerant computer apparatus for use in systems, the apparatus comprising:

- a plurality of processes executing on at least one processor (col. 6, lines 33-48);

- at least one application executing in a pure fault tolerant mode where said application is in an active condition on one said process and in a standby condition on another said process on said processors (col. 6, lines 33-48 and col. 7, lines 10-34);
- a router for providing communications between at least one said application and other applications independent of application locations (col. 6, lines 60-65); and
- an ADSM for providing fault tolerant functionality in said application and wherein said application is represented by a single resource set (col. 4, lines 22-30 and col. 8, lines 35-63).

However, Wyld does not explicitly disclose:

- a system controller for controlling system activation and failure recovery.

Rathunde discloses a system and method for performing dynamic load balancing during system operation between a pair of application processes including:

- a system controller for controlling system activation and failure recovery (col. 11, lines 41-59).

Given the teaching of Rathunde, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify Wyld incorporating or implementing a failure recovery technique in order to improve performance and reliability of the processors by controlling the failure of processors on the network.

As per claim 48, Wyld discloses a distributed processing, fault tolerant computer apparatus for use in systems, the apparatus comprising:

- a plurality of processes executing on at least one processor (col. 6, lines 33-48);

- at least one application executing in a distributed fault tolerant mode where said application is in an active condition on more than one of said processes and is in a standby condition on at least one of said processes on said processors (col. 6, lines 33-48 and col. 7, lines 10-34);
- a router for providing communications between at least one said application and other applications independent of application locations (col. 6, lines 60-65).
- an ADSM for providing distributed fault tolerant functionality in said application (col. 8, lines 35-63); and
- an ALDM for distributing incoming events to said application (col. 8, lines 35-63).

However, Allen does not explicitly disclose:

- a system controller for controlling system activation, failure recovery and initial load distribution.

Rathunde discloses a system and method for performing dynamic load balancing during system operation between a pair of application processes including:

- a system controller for controlling system activation, failure recovery and initial load distribution (col. 8, lines 17-32, col. 9, lines 19-26 and lines 37-48).

Given the teaching of Rathunde, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify Wyld incorporating or implementing a load balancing technique in order to improve the load sharing among the application processors by balancing incoming messages/packets across the processors.

As per claims **2** and **49**, Wyld discloses the invention substantially as claimed.

However, Wyld does not explicitly disclose:

- wherein said system controller also provides procedures for controlling any one or more members of the group consisting of fault recovery, load redistribution, system topology, and system maintenance.

Rathunde discloses a system and method for performing dynamic load balancing during system operation between a pair of application processes including:

- wherein said system controller also provides procedures for controlling any one or more members of the group consisting of fault recovery, load redistribution, system topology, and system maintenance (col. 8, lines 17-32, col. 9, lines 19-26 and lines 37-48).

Given the teaching of Rathunde, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify Wyld incorporating or implementing a load balancing technique in order to improve the load sharing among the application processors by balancing incoming messages/packets across the processors.

As per claims **3** and **50**, Wyld further discloses:

- a plurality of resource sets each being a unit of distribution, and said application using more than one said resource set (col. 4, lines 22-30 and col. 8, lines 35-63).

As per claim **4**, Wyld discloses:

- wherein shared data in said application is modified by a master critical resource set and updated onto shadow resource sets on all copies of said application and private data in said application is modified by active non-critical resource sets (col. 4, lines 22-30 and col. 8, lines 35-63).

As per claims **5**, **30**, and **52**, Wyld discloses:

- wherein said ADSM provides API for making a resource set active (col. 4, lines 22-30 and col. 8, lines 35-63).

As per claims **6**, **31**, and **53**, Wyld discloses:

- wherein said ADSM provides API for making a resource set standby and to warm start said standby resource set (col. 4, lines 22-30 and col. 8, lines 35-63).

As per claims **7**, **32**, and **54**, Wyld discloses:

- wherein said ADSM provides API for making a resource set out of service (col. 4, lines 22-30 and col. 8, lines 35-63).

As per claims **8**, **33**, and **55**, Wyld discloses:

- wherein said ADSM provides API to disable peer update towards a resource set (col. 4, lines 22-30 and col. 8, lines 35-63).

As per claim **51**, Wyld discloses:

- wherein shared data in said application is modified by a master critical resource set and updated onto shadow resource sets on all copies of said application and private data in said application is modified by active non-critical resource sets and updated onto standby resource sets (col. 4, lines 22-30 and col. 6, lines 44-65).

As per claims **9** and **56**, Wyld discloses the invention substantially as claimed.

However, Wyld does not explicitly disclose:

- wherein said ALDM distributes the processing load by mapping incoming events to said resource sets and sending events to said active resource set.

Rathunde discloses a system and method for performing dynamic load balancing during system operation between a pair of application processes including:

- wherein said ALDM distributes the processing load by mapping incoming events to said resource sets and sending events to said active resource set (col. 8, lines 17-32, col. 9, lines 19-26 and lines 37-48).

Given the teaching of Rathunde, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify Wyld incorporating or implementing a load balancing technique in order to improve the load sharing among the application processors by balancing incoming messages/packets across the processors.

As per claims **10** and **57**, Wyld discloses:

- wherein said ALDM provides API to set the weight of a resource set (col. 4, lines 22-30 and col. 8, lines 35-63).

As per claims **11** and **58**, Wyld discloses:

- APIs of said ALDM (col. 4, lines 22-30 and col. 8, lines 35-63),
- APIs of said ADSM (col. 4, lines 22-30 and col. 8, lines 35-63),
- APIs of said router (col. 6, lines 60-65), and
- APIs of said system controller (col. 6, lines 60-65).

As per claim **29**, Wyld discloses:

- wherein data in said application is modified by a single active resource set and updated on a standby resource set (col. 4, lines 22-30 and col. 6, lines 44-65).

As per claims **12**, **34**, and **59**, Wyld discloses:

- wherein said router provides API to send messages to said active resource set of said application (col. 4, lines 22-30 and col. 6, lines 44-65).

As per claims **13**, **35**, and **60**, Wyld discloses:

- wherein said router provides API to set and clear active mapping for said resource sets (col. 4, lines 22-30 and col. 6, lines 60-65).

As per claims **14**, **36**, and **61**, Wyld discloses:

- wherein said router provides API to set and clear standby mapping for said resource sets (col. 4, lines 22-30 and col. 6, lines 44-65).

As per claims **15** and **62**, Wyld discloses:

- wherein said router provides API to set and clear master mapping for said master critical resource set and to add and remove shadow mapping from a multicast list for said critical resource set (col. 4, lines 22-30 and col. 6, lines 44-65).

As per claims **16**, **37**, and **63**, Wyld discloses:

- wherein said router provides API to hold and release messages for said resource sets (col. 4, lines 22-30 and col. 6, lines 44-65).

As per claims **17**, **38**, and **64**, Wyld discloses:

- wherein said router provides API to perform adjacent ping for flushing communication channels and to peer sync messages held for said resource sets with said router (col. 4, lines 22-30, col. 5, lines 12-21 and col. 6, lines 44-65).

As per claims **18**, **39**, and **65**, Allen discloses:

- wherein said router provides API to send update messages to a standby resource set (col. 4, lines 22-30, col. 5, lines 12-21 and col. 6, lines 44-65).

As per claims **19** and **66**, Wyld discloses:

- wherein said router provides API to send messages to all said shadows in a multicast list of said critical resource set (col. 4, lines 22-30, col. 5, lines 12-21 and col. 6, lines 44-65).

As per claims **20** and **67**, Wyld discloses:

- wherein said system controller is configured with all of the said applications in the system, with mode of operation for each said application, said critical and non-critical resource sets information of each said application and service user/provider relationship between said applications (col. 4, lines 22-30, col. 5, lines 12-21 and col. 6, lines 44-65).

As per claims **21**, **41**, and **68**, Wyld discloses:

- wherein said system controller provides resource set level API to make a resource set active (col. 4, lines 22-30, col. 5, lines 12-21 and col. 6, lines 44-65)

As per claims **22**, **42**, and **69**, Wyld discloses:

- wherein said system controller provides resource set level API to make a resource set standby (col. 4, lines 8-22, col. 5,lines 1-27, and col. 6, lines 10-42).

As per claims **23**, **43**, and **70**, Wyld discloses:

- wherein said system controller provides resource set level API to make a resource set out of service (col. 2, lines 61-67, col. 3, lines 1-4, col. 4, lines 8-22, col. 5,lines 1-27, and col. 6, lines 10-42).

As per claims **24**, **44**, and **71**, Wyld discloses:

- wherein said system controller provides resource set level API to perform any one or more of the group consisting of forced switchover, controlled switchover, forced move and controlled move operation (col. 2, lines 53-67, and col. 3, lines 1-5).

As per claims **25**, **45**, and **72**, Wyld discloses the invention substantially as claimed. However, Wyld does not explicitly disclose:

- wherein said system controller provides application level enable node API to introduce a process with at least one application into a system during initialization, for scaling an operational system, and wherein said system controller implements algorithms to redistribute the load between all said processes by movement of resource sets.

Rathunde discloses a system and method for performing dynamic load balancing during system operation between a pair of application processes including:

- wherein said system controller provides application level enable node API to introduce a process with at least one application into a system during initialization, for scaling an operational system, and wherein said system controller implements algorithms to redistribute the load between all said processes by movement of resource sets (col. 8, lines 17-32, col. 9, lines 19-26 and lines 37-48).

Given the teaching of Rathunde, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify Wyld incorporating or implementing a load balancing technique in order to improve the load sharing among the application processors by balancing incoming messages/packets across the processors.

As per claims **26**, **46**, and **73**, Wyld discloses the invention substantially as claimed. However, Wyld does not explicitly disclose:

- wherein said system controller provides application level disable node API to recover from the failure of at least one application in a process and wherein said system controller redistributes the load by movement of resource sets.

Rathunde discloses a system and method for performing dynamic load balancing during system operation between a pair of application processes including:

- wherein said system controller provides application level disable node API to recover from the failure of at least one application in a process and wherein said system controller redistributes the load by movement of resource sets (col. 8, lines 17-32, col. 9, lines 19-26 and lines 37-48).

Given the teaching of Rathunde, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify Wyld incorporating or implementing a load balancing technique in order to improve the load sharing among the application processors by balancing incoming messages/packets across the processors.

As per claims **27**, **47** and **74**, Wyld discloses the invention substantially as claimed.

However, Wyld does not explicitly disclose:

- wherein said system controller provides application level disable node API to shutdown at least one said application in a process and wherein said system controller redistributes the load by movement of resource sets.

Rathunde discloses a system and method for performing dynamic load balancing during system operation between a pair of application processes including:

- wherein said system controller provides application level disable node API to shutdown at least one said application in a process and wherein said system controller redistributes

the load by movement of resource sets (col. 8, lines 17-32, col. 9, lines 19-26 and lines 37-48).

Given the teaching of Rathunde, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify Wyld incorporating or implementing a load balancing technique in order to improve the load sharing among the application processors by balancing incoming messages/packets across the processors.

As per claim **40**, Wyld discloses:

- wherein said system controller is configured with all of the said applications in the system, with mode of operation for each said application, and service user/provider relationship between said applications (col. 2, lines 61-67, col. 3, lines 1-4, col. 4, lines 8-22, col. 5,lines 1-27, and col. 6, lines 10-42).

As per claim **75**, Wyld discloses a distributed processing, computer apparatus for use in systems, the apparatus comprising:

- a plurality of processes executing on at least one processor (col. 6, lines 33-48);
- at least one application executing in a pure distributed mode where said application is distributed in an active condition among more than one of said processes on said processors (col. 6, lines 33-48 and col. 7, lines 10-34);
- a router for providing communications between at least one said application and other applications independent of application locations (col. 6, lines 60-65); and
- an update module for providing distributed functionality in said application (col. 4, lines 22-30 and col. 6, lines 44-65).

However, Allen does not explicitly disclose:

- a system controller for controlling system activation and initial load distribution; and
- load distributor for distributing incoming events to said application.

Rathunde discloses a system and method for performing dynamic load balancing during system operation between a pair of application processes including:

- a system controller for controlling system activation and initial load distribution (col. 8, lines 17-32, col. 9, lines 19-26 and lines 37-48); and
- load distributor for distributing incoming events to said application (col. 8, lines 17-32, col. 9, lines 19-26 and lines 37-48).

Given the teaching of Rathunde, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify Wyld incorporating or implementing a load balancing technique in order to improve the load sharing among the application processors by balancing incoming messages/packets across the processors.

As per claim **76**, Wyld discloses a fault tolerant computer apparatus for use in systems, the apparatus comprising:

- a plurality of processes executing on at least one processor (col. 6, lines 33-48);
- at least one application executing in a pure fault tolerant mode where said application is in an active condition on one said process and in a standby condition on another said process on said processors (col. 6, lines 33-48 and col. 7, lines 10-34);
- a router for providing communications between at least one said application and other applications independent of application locations (col. 6, lines 60-65); and

- an update module for providing fault tolerant functionality in said application and wherein said application is represented by a single reserved resource set (col. 4, lines 22-30 and col. 6, lines 44-65).

However, Allen does not explicitly disclose:

- a system controller for controlling system activation and failure recovery.

Rathunde discloses a system and method for performing dynamic load balancing during system operation between a pair of application processes including:

- a system controller for controlling system activation and failure recovery (col. 11, lines 41-59).

Given the teaching of Rathunde, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify Wyld incorporating or implementing a failure recovery technique in order to improve performance and reliability of the processors by controlling the failure of processors on the network.

As per claim **77**, Wyld discloses a distributed processing, fault tolerant computer apparatus for use in systems, the apparatus comprising:

- a plurality of processes executing at least one processor; at least one application executing in a distributed .tolerant mode where said application is in an active condition on more than one of said processes and is in a standby condition on at least one of said processes on said processors (abstract, col. 4, lines 8-22, and col. 6, lines 10-42);
- a router for providing communications between at least one said application and other applications independent of application locations (col. 5,lines 1-27); and

- an update module for providing distributed fault tolerant functionality in said application.

However, Wyld does not disclose:

- a system controller for controlling system activation, failure recovery and initial load distribution; and
- a load distributor for distributing incoming events to said application.

Rathunde discloses a system and method for performing dynamic load balancing during system operation between a pair of application processes including:

- a system controller for controlling system activation, failure recovery and initial load distribution (col. 8, lines 17-32, col. 9, lines 19-26 and lines 37-48); and
- a load distributor for distributing incoming events to said application (col. 8, lines 17-32, col. 9, lines 19-26 and lines 37-48).

Given the teaching of Rathunde, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify Wyld incorporating or implementing a load balancing technique in order to improve the load sharing among the application processors by balancing incoming messages/packets across the processors.

As per claims **82** and **90**, Wyld discloses the invention substantially as claimed.

However, Wyld does not explicitly disclose:

- distributing the processing load of said application by mapping incoming events to said resource sets of said application and sending events to active resource sets.

Rathunde discloses a system and method for performing dynamic load balancing during system operation between a pair of application processes including:

- distributing the processing load of said application by mapping incoming events to said resource sets of said application and sending events to active resource sets (col. 8, lines 17-32, col. 9, lines 19-26 and lines 37-48).

Given the teaching of Rathunde, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify Wyld incorporating or implementing a load balancing technique in order to improve the load sharing among the application processors by balancing incoming messages/packets across the processors.

As per claims **86** and **93**, Wyld discloses the invention substantially as claimed.

However, Wyld does not explicitly disclose:

- dynamic load balancing by either moving the resource sets from one said process to other said process or by mapping new events to relatively idle resource sets.

Rathunde discloses a system and method for performing dynamic load balancing during system operation between a pair of application processes including:

- dynamic load balancing by either moving the resource sets from one said process to other said process or by mapping new events to relatively idle resource sets (col. 8, lines 17-32, col. 9, lines 19-26 and lines 37-48).

Given the teaching of Rathunde, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify Wyld incorporating or implementing a load balancing technique in order to improve the load sharing among the application processors by balancing incoming messages/packets across the processors.

Response to Arguments

5. Applicant's arguments with respect to claims 1-99 have been considered but are moot in view of the new ground(s) of rejection.

Conclusion

6. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

U.S. Pat. No. 6,427,213 to Dao

U.S. Pat. No. 6,223,304 to Kling et al

U.S. Pat. No. 6,434,712 to Urban et al

U.S. Pat. No. 6,629,263 to Sassi

Any inquiry concerning this communication or earlier communications from the examiner should be directed to LaShonda T. Jacobs whose telephone number is 703-305-7494. The examiner can normally be reached on 8:30 AM - 5:00 PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Ario Etienne can be reached on 703-308-7562. The fax phone number for the organization where this application or proceeding is assigned is (703) 872-9306.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the receptionist whose telephone number is 703-305-3900.

LaShonda T. Jacobs
Examiner
Art Unit 2157

ltj
April 29, 2004

ARIO ETIENNE
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100